

## Claims

What is claimed is:

- [c1] A network system for key management, comprising:
- a server;
  - a key management system providing process logic for key management system initialization located on the server;
  - a key management system storage providing a secure data storage for the key management system; and
  - an interface providing a means for inputting data into the key management system.
- [c2] The network system of claim 1, further comprising a client computer operatively connected to the server, wherein the client computer comprises a user interface to input data into the key management system.
- [c3] The network system of claim 1, wherein the key management storage is located on the server.
- [c4] The network system of claim 1, wherein the key management storage is located on a second server operatively connected to the server.
- [c5] The network system of claim 1, wherein the interface comprises a graphical user interface.
- [c6] The network system of claim 5, wherein the graphical user interface is integrated into a web browser.
- [c7] The network system of claim 2, wherein the user interface comprises a graphical user interface.

- [c8] The network system of claim 7, wherein the graphical user interface is integrated into a web browser.
- [c9] The network system of claim 2, wherein the client computer and the server are connected using an encrypted connection.
- [c10] The network system of claim 1, wherein the key management system further comprises:  
a memory storing data within the key management system;  
a hashing module hashing a key encryption key;  
an encryption module encrypting data; and  
a serialization module serializing data obtained from the memory, the encryption module, and the serialization module.
- [c11] The key management system of claim 10, further comprising:  
a randomizer randomizing data.
- [c12] The key management system of claim 10, further comprising:  
an encoding module for encoding data.
- [c13] The key management system of claim 10, wherein the hashing module uses a MD5 hashing function.
- [c14] The key management system of claim 10, wherein the encryption module, further comprises a key generation tool.
- [c15] The key management system of claim 14, wherein the key generation tool comprises a symmetric algorithm.
- [c16] The key management system of claim 14, wherein the key generation tool comprises an asymmetric algorithm.

- [c17] A network system for key management, comprising:
- a server;
  - a key management system providing process logic for key management system initialization located on the server;
  - a key management system storage providing a secure data storage for the key management system;
  - an interface providing a means for inputting data into the key management system;
  - and
  - a client computer operatively connected to the server, wherein the client computer comprises a user interface to input data into the key management system.
- [c18] A method for initializing a key management system comprising:
- entering data into a key management system interface;
  - entering a key encryption key into the key management system interface;
  - combining data into a tuple;
  - encrypting the tuple with the key encryption key to produce a secret token;
  - storing the secret token in a vector;
  - hashing the key encryption key;
  - storing a hashed key encryption key in the vector;
  - storing a list of keys in the vector;
  - serializing the vector to produce a serialized file; and
  - storing the serialized file in a key management system storage.
- [c19] The method of claim 18, further comprising:
- encoding a key field of the tuple.
- [c20] The method of claim 19, further comprising:
- randomizing the order of the list of encoded keys.

- [c21] The method of claim 18, further comprising:  
randomizing the order of the secret tokens in the vector.
- [c22] The method of claim 18, further comprising:  
randomizing the order of the list of keys.
- [c23] The method of claim 18, further comprising:  
generating data to encrypt;
- [c24] The method of claim 18, wherein the tuple comprises:  
a key field;  
a value field; and  
a type field.
- [c25] The method of claim 18, wherein the tuple comprises:  
an application name field;  
a key field;  
a value field; and  
a type field.
- [c26] The method of claim 18, wherein the vector comprises:  
a secret token portion;  
a key encryption key hash portion; and  
a key list portion.
- [c27] The method of claim 26, further comprising;  
tagging the secret token with an application name.
- [c28] The method of claim 26, further comprising:  
tagging the key in the key list with an application name.

- [c29] The method of claim 18, wherein the key management storage is located on a second server.
- [c30] The method of claim 18, wherein the key management system interface comprises a graphical user interface.
- [c31] The method of claim 30, wherein the graphical user interface is integrated into a web browser.
- [c32] The method of claim 18, wherein the encrypting comprises using a symmetric algorithm.
- [c33] The method of claim 18, wherein the encrypting comprises using an asymmetric algorithm.
- [c34] A method for initializing a key management system comprising:  
entering data into a key management system interface;  
entering a key encryption key into the key management system interface;  
combining data into a tuple;  
encrypting the tuple with the key encryption key to produce a secret token;  
storing the secret token in a vector;  
hashing the key encryption key;  
storing a hashed key encryption key in the vector;  
storing a list of keys in the vector;  
serializing the vector to produce a serialized file;  
storing the serialized file in a key management system storage;  
encoding a key field of the tuple;  
randomizing the order of the list of keys;  
randomizing the order of the secret tokens in the vector; and  
generating data to encrypt.

- [c35] An apparatus for initializing a key management system comprising:
- means for entering data into a key management system interface;
  - means for entering a key encryption key into the key management system interface;
  - means for combining data into a tuple;
  - means for encrypting the tuple with the key encryption key to produce a secret token;
  - means for storing the secret token in a vector;
  - means for hashing the key encryption key;
  - means for storing a hashed key encryption key in the vector;
  - means for storing a list of keys in the vector;
  - means for serializing the vector to produce a serialized file;
  - means for storing the serialized file in a key management system storage;
  - means for encoding a key field of the tuple;
  - means for randomizing the order of the list of keys;
  - means for randomizing the order of the secret tokens in the vector; and
  - means for generating data to encrypt.